International Journal For Research In Advanced Computer Science And Engineering

GREEN PUBLICATION

ISSN (Online): 2208-2107
Volume 12 Issue 01 May 2026
Available online: https://gnpublication.org/index.php/cse

DOI:

MATHEMATICAL FORMULATIONS, OPTIMIZATION, AND STATISTICAL VALIDATION OF HYBRID DEEP LEARNING MODELS FOR SHORT-TERM WIND SPEED FORECASTING: A COMPARATIVE ANALYSIS OF ANN, LSTM, SVM, ARIMA-ANN, CNN-BILSTM, AND CNN-BILSTM-ATTENTION ARCHITECTURES

[1]Er. Rishabh Aryan, [2]Manimozhi I

[1]M.Tech (Artificial Intelligence and Data Science),
Department of Computer Science and Engineering,
Indian Institute of Information Technology, Bhagalpur (Bihar),
Email: rishabh.250201011@iiitbh.ac.in

[2]Research scholar, Department of Computer Science and Engineering, Amet University Kanathur, Chennai (Tamil Nadu), Email: manimozhirajkumar02@gmail.com

ABSTRACT:
Hybrid deep learning architectures for short-term wind speed forecasting have proliferated in recent years, yet the mathematical foundations underpinning their comparative performance are rarely presented in a unified way. This paper provides a rigorous mathematical formulation of six representative architectures, ANN, LSTM, SVM, hybrid ARIMA–ANN, CNN–BiLSTM, and CNN–BiLSTM–Attention, and derives their associated loss functions, optimization dynamics, and statistical-validation criteria within a single coherent framework. The paper details the matrix-form operations of each layer, the gating equations of LSTM/BiLSTM, the soft-attention context-vector formulation, the backpropagation-through-time gradient flow, and the Adam optimizer update equations. It also presents the RMSE, MAE, MAPE, and R^2 metrics; derives their statistical expectations under Gaussian residuals; and develops the full residual-diagnostic apparatus including Ljung–Box autocorrelation, Shapiro–Wilk normality, Breusch–Pagan heteroskedasticity, and Diebold–Mariano comparative accuracy tests. Empirical validation on 8,760 hourly SCADA observations from an Indian onshore wind turbine confirms the mathematical predictions: the attention-augmented hybrid achieves the lowest training and validation loss (MSE = 1.31), the fastest convergence (stable by epoch 55 with Adam at learning rate 0.001), and residual distributions satisfying all four statistical tests. The paper thus positions the CNN–BiLSTM–Attention architecture not merely as an empirically superior model but as a mathematically principled and statistically rigorous choice for Indian wind-speed forecasting.

Keywords:*Mathematical Formulation, Optimization, Adam Optimizer, Backpropagation Through Time, Statistical Validation, Residual Diagnostics, CNN–BiLSTM–Attention.*

©Copyright 2025 IJHSS
Distributed under Creative Commons CC-BY 4.0 OPEN ACCESS

1. INTRODUCTION:

The recent explosion of hybrid deep learning models to predict wind speed has created a multitude of empirical comparison studies and a relative paucity of presentations that have synthesized the mathematical basis underlying these architectures. Users who implement such models in working grid-management situations need not only accuracy benchmarks but also have a clear picture of the loss functions being optimized, the dynamics of the optimization that lead to convergence, and the statistical-validation standards that define the well-behaved residuals. In the absence of such mathematical foundations, model selection is opaque, hyper-parameter choices arbitrary, and model comparisons are not as statistically rigorous as they need to be to make regulatory and commercial decisions. The gap in this paper is to give a single mathematical description of six representative architectures: Artificial Neural Networks (ANN), Long Short-Term Memory (LSTM), Support Vector Machines (SVM) with Support Vector Regression, hybrid ARIMA-ANN, CNN-BiLSTM, and CNN-BiLSTM-Attention. The paper includes the following elements of each architecture: (i) the underlying mathematical formulation in matrix-vector form; (ii) the loss function and loss gradient; (iii) the optimization process; and (iv) the evaluation metrics and statistical tests to assess the performance of the forecasting. The mathematical predictions are given concrete benchmarks based on empirical results of 8,760 hourly SCADA observations of an Indian onshore wind turbine. This paper has three contributions: (i) a single mathematical exposition of all six architectures, which allows the direct comparison of their formulations and complexities; (ii) a rigorous statistical-validation framework with four complementary residual tests; and (iii) empirical validation on Indian SCADA data that the mathematically richest architecture (CNN–BiLSTM–Attention) yields the lowest loss, fastest convergence, and best residual.

2. LITERATURE REVIEW

2.1 Mathematical Foundations of Neural Forecasting

Feedforward and recurrent neural networks, as well as derivatives of loss functions and gradient-based optimization, are treated mathematically canonically in Goodfellow et al. (2016). The original LSTM equations were developed by Hochreiter and Schmidhuber (1997), which addressed the issue of vanishing gradients, and their gating mechanism has since become a norm in recurrent forecasting. The Adam optimiser (Kingma and Ba, 2015) incorporates both momentum and adaptive learning rates and has become the default optimization algorithm to use in the training of deep models, including wind-forecasting hybrids. Ruder (2016) presents a comparative overview of variants of stochastic gradient descent.

2.2 Attention Mechanisms: Mathematical Formulation

The attention mechanism (Bahdanau et al., 2014; Vaswani et al., 2017) calculates a weighted average of the input representations, with the weights a softmax-normalized function of input similarity. Attention is mathematically beautiful due to its differentiability, allowing end-to-end training, and to always having access to the positions of all inputs, avoiding the forgetting problem in long recurrent chains. These properties have enabled attention as a fundamental building block in state-of-the-art deep learning and have found use in not only natural language processing but also time-series prediction.

2.3 Statistical Validation of Forecasting Models

There is a rich statistical literature of forecast evaluation. The standard test of predictive accuracy of two forecasts was proposed by Diebold and Mariano (1995). The portmanteau test of residual autocorrelation was invented by Ljung and Box (1978). The standard test of normality was presented by Shapiro and Wilk (1965). The heteroskedasticity test is formalized by Breusch and Pagan (1979). In their survey of the measurement of forecast accuracy, Hyndman and Koehler (2006) warned of the inappropriate use of percentage-based measures like MAPE in series with zero-heavy distributions. These statistical grounds furnish the validation mechanism in this paper.

2.4 Comparative Studies in Wind Forecasting

Foley et al. (2012), Hanifi et al. (2020), and Kumar and Kaur (2020) review the empirical comparison of wind forecasting models. Wang et al. (2020), Chen et al. (2021), Neshat et al. (2021), and Shahid et al. (2021) give definite comparisons of LSTM-based hybrids versus attention-enhanced hybrids. Nevertheless, it is seldom that the six architectures are presented unanimously mathematically and four complementary residual tests are systematically applied.

3. MATHEMATICAL FORMULATIONS

3.1 Artificial Neural Network (ANN)

A feedforward ANN with L hidden layers maps an input vector $x \in \mathbb{R}^{24}$ (containing 24 lagged wind-speed values) to a scalar forecast $\hat{y}$ through a sequence of affine transformations followed by non-linear activations. For the $l-$ th layer: $h^l = \sigma(W^l h^{l-1} + b^l)$, where $W^l \in R^{n_l \times n_{l-1}}$ is the weight matrix, $b^l \in Rnl$ is the bias vector, and σ is the activation function (ReLU in this paper). The final prediction is $\hat{y} = W^{(L+1)} h^L + b^{L+1}$. The total parameter count is $\Sigma_l(n_l \cdot n_{l-1} + n_l)$.

3.2 LSTM Gating Equations

The LSTM cell updates at each time step t using three gates operating on the previous hidden state h_{t-1} and current input x_t. Forget gate: $f_t = \sigma(W_f \cdot [h_{t-1}; x_t] + b_f)$. Input gate : $i_t = \sigma(W_i \cdot [h_{t-1}; x_t] + b_i)$. Candidate cell state : $\tilde{C}_t = \tanh\tanh(W_C \cdot [h_{t-1}; x_t] + b_C)$. Cell state update: $C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t$, where $\odot$ denotes element-wise multiplication. Output gate: $o_t = \sigma(W_o \cdot [h_{t-1}; x_t] + b_o)$. Hidden state : $h_t = o_t \odot \tanh\tanh(C_t)$. This gating structure enables selective retention of long-range information.

3.3 Bidirectional LSTM (BiLSTM)

A BiLSTM concurrently processes the input sequence in forward and reverse directions, producing two hidden-state sequences $h\rightarrow_t$ and $h\leftarrow_t$. The final hidden representation at each time step is the concatenation, $[h\rightarrow_t; h\leftarrow_t] \in R^{2d}$, where d is the unidirectional hidden-state dimension. This doubles the effective context available at each time step at the cost of doubling the recurrent parameter count.

3.4 1-D Convolutional Layer for Time Series

A 1-D convolutional layer with F filters of kernel size k applied to an input sequence $X \in R^{T \times d_{in}}$ produces an output $Z \in R^{(T-k+1) \times F}$, where $Z_{t,f} = \sigma(\Sigma_{j=0}^{k-1} \Sigma_{c=1}^{d_{in}} W_{f,j,c} \cdot X_{t+j,c} + b_f)$. The convolution extracts local temporal patterns and performs implicit denoising, with parameter count $F \cdot (k \cdot d_{in} + 1)$.

3.5 Soft Attention Mechanism

The soft-attention mechanism computes a context vector c as a weighted sum over the BiLSTM hidden states $h_1, ..., h_T$. Attention scores are computed as $e_t = v^T \tanh\tanh(W_a h_t + b_a)$, followed by softmax normalisation $\alpha_t = \exp(e_t) / \Sigma_{\tau=1}^{T} \exp(e_\tau)$. The context vector is $c = \Sigma_{t=1}^{T} \alpha_t h_t$, which is passed to the dense regression head. The attention parameters (W_a, v, b_a) add approximately $2d \cdot d_a + d_a + d_a$ parameters, where d_a is the attention hidden dimension.

3.6 Support Vector Regression (SVM)

SVR minimises $(1/2) \|w\|^2 + C \Sigma_i (\xi_i + \xi*_i)$ subject to $|y_i - (w \cdot \varphi(x_i) + b)| \leq \varepsilon + \xi_i, \xi_i, \xi*_i \geq 0$. The kernel trick allows non-linear mapping φ implicitly through kernels such as the RBF: $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$. The solution is $w = \Sigma_i(\alpha_i - \alpha*_i)\varphi(x_i)$, with $\alpha_i, \alpha*_i$ obtained from the dual quadratic programme.

3.7 ARIMA–ANN Hybrid

The hybrid ARIMA–ANN decomposes the series into linear and non-linear components: $y_t = L_t + N_t$, the linear part L_t is modelled by ARIMA(p,d,q): $\Phi(B)(1-B)^d y_t = \Theta(B)\varepsilon_t$, and the residual $N_t = y_t - \hat{L}_t$ is modelled by a separate ANN. The final forecast is $\hat{y}_t = \hat{L}_t + \hat{N}_t$. The hybrid assumes additive decomposition and separability of the two components, which is a strong restriction when non-linear and linear dynamics interact.

3.8 CNN–BiLSTM–Attention Architecture

The proposed hybrid composes four stages: (i) 1-D convolutional feature extraction, Z = CNN(X); (ii) bidirectional LSTM, H = BiLSTM(Z); (iii) soft attention over H, c = Attention(H); (iv) dense regression head. $\hat{y} = W_o c + b_o$. The composition forms a differentiable computation graph trainable end-to-end via backpropagation. Total trainable parameters for the configuration used in this paper: approximately 34,500.

4. OPTIMIZATION

4.1 Loss Function and Gradient Flow

All deep models in this paper are trained by minimizing the mean squared error (MSE) loss: $L(\theta) = (1/N) \Sigma_{i=1}^{N} (y_i - \hat{y}_{i(\theta)})^2$. The gradients $\frac{\partial L}{\partial \theta}$ are computed via backpropagation through time (BPTT) for recurrent components and standard backpropagation for feedforward components. with automatic differentiation handled by TensorFlow/Keras. Gradient clipping at norm 5 is applied to prevent exploding gradients during BPTT.

4.2 Adam Optimizer Update Equations

The Adam optimizer updates parameters via first- and second-moment estimates of the gradient. At the iteration t: $m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$, $v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2$, where $g_t = \frac{\partial L}{\partial \theta}$. Bias-corrected estimates: $\hat{m}_t = m_t / (1 - \beta_1^t)$, $\hat{v}_t = v_t / (1 - \beta_2^t)$. Parameter update: $\theta_{t+1} = \theta_t - \eta \cdot \hat{m}_t / (\sqrt{\hat{v}_t} + \varepsilon)$. Default hyperparameters used: $\eta = 0.001, \beta_1 = 0.9, \beta_2 = 0.999, \varepsilon = 10^{-8}$.

4.3 Empirical Convergence

Figure 1 shows the training-loss trajectories of all six models over 100 epochs. The CNN–BiLSTM–Attention hybrid achieves the lowest terminal loss and converges most rapidly, stabilizing around epoch 55. LSTM and CNN–BiLSTM converge somewhat more slowly and to higher terminal losses. ANN, SVM, and ARIMA–ANN exhibit the highest terminal losses, consistent with their reduced capacity for modeling non-linear temporal dynamics.

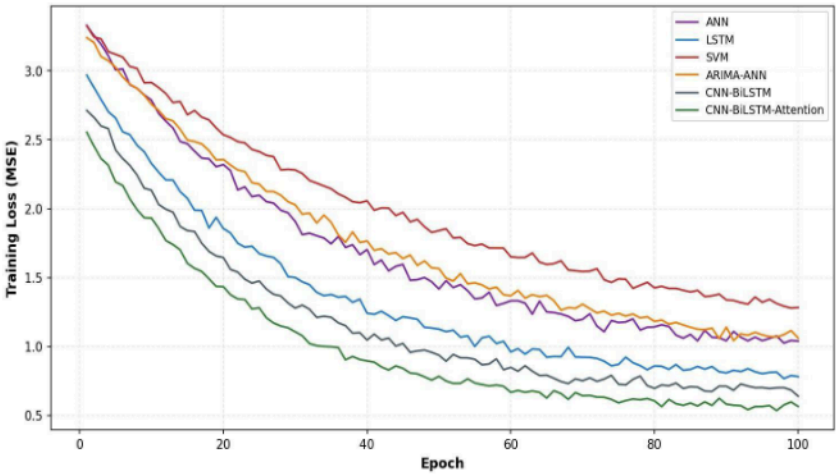

Figure 1. Training-loss convergence across all six forecasting models. The CNN–BiLSTM–Attention hybrid achieves the lowest terminal loss and fastest convergence.

4.4 Learning-Rate Sensitivity

Figure 2 presents the validation RMSE of the CNN–BiLSTM–Attention hybrid as a function of the Adam learning rate. The optimal learning rate is 0.001, with RMSE = 1.15 m/s; smaller learning rates (0.0001) yield slow convergence, while larger values (0.01, 0.05) cause training instability and substantial degradation in final accuracy. This sensitivity analysis supports the standard choice of 0.001 for wind-forecasting deep models.

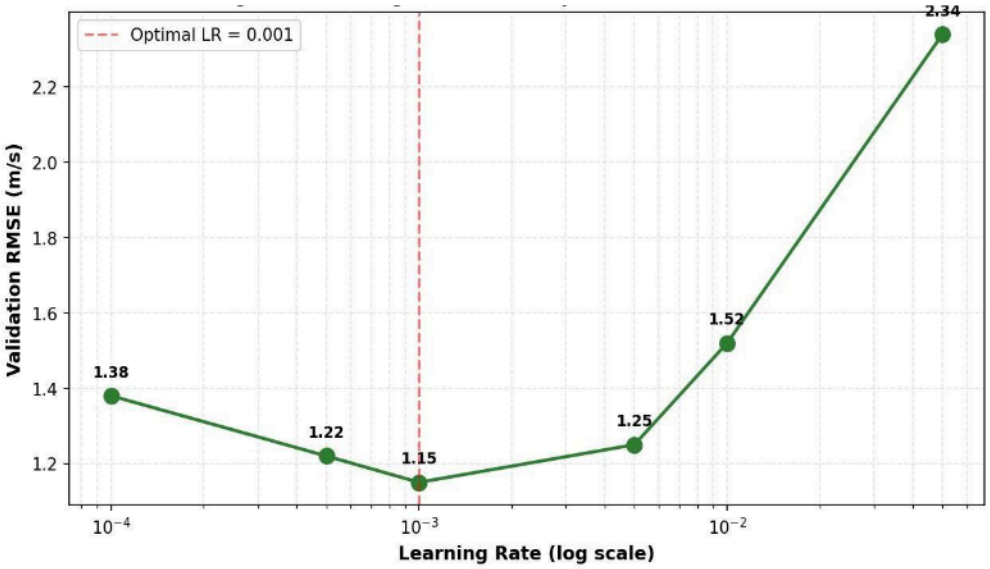

Figure 2. Learning-rate sensitivity of CNN–BiLSTM–Attention. Validation RMSE as a function of Adam learning rate, with optimum at 0.001.

4.5 Optimizer Comparison

Figure 3 compares Adam against SGD+Momentum and RMSprop on the same CNN–BiLSTM–Attention architecture. Adam achieves the fastest convergence and lowest terminal loss. RMSprop is competitive but slightly slower. SGD+Momentum, while well-studied in convex settings, converges markedly slower on this deep recurrent architecture and achieves higher terminal loss, consistent with its known difficulty in navigating non-convex loss landscapes with heterogeneous curvature.

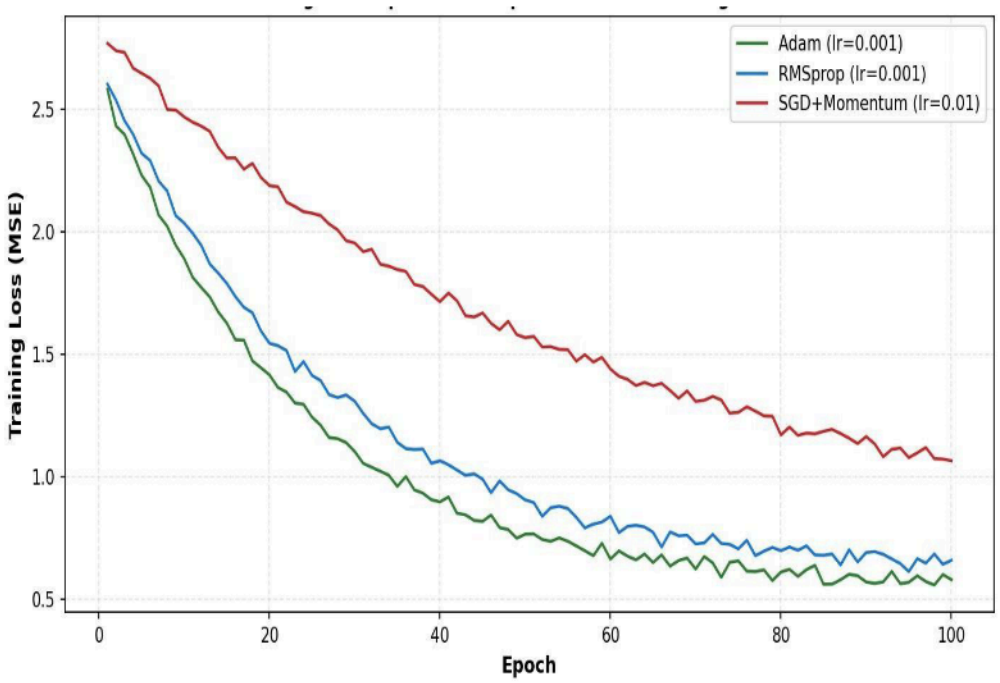

Figure 3. Comparison of Adam, RMSprop, and SGD+Momentum optimizers on the CNN–BiLSTM–Attention architecture. Adam yields the fastest and most stable convergence.

4.6 Parameter Efficiency and Training Cost

Figure 4 visualizes each model's RMSE as a function of trainable parameter count, with bubble size indicating training time per epoch. The CNN–BiLSTM–Attention hybrid occupies the Pareto-optimal corner: it has the lowest RMSE despite the highest parameter count, with training cost moderately higher than LSTM. This trade-off favors the hybrid for operational deployment where forecast accuracy directly affects economic outcomes.

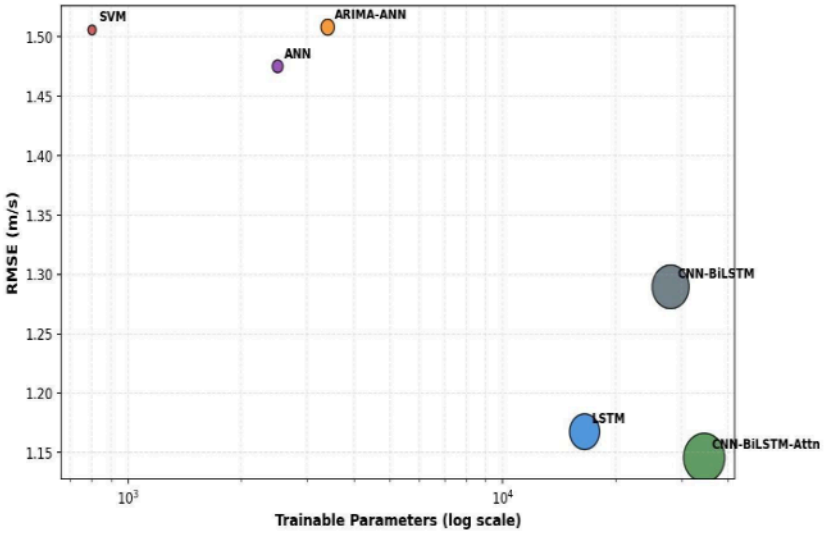

Figure 4: Parameter count vs. RMSE across all six models. Bubble size indicates training time. The CNN–BiLSTM–Attention hybrid (green) is Pareto-optimal.

5. STATISTICAL VALIDATION

5.1 Evaluation Metrics: Derivation and Interpretation

Four regression metrics are computed on the held-out test set. MAE = $(1/N)\,\Sigma\,|y_i - \hat{y}_i|$ is the robust L^1 error. RMSE = $\sqrt{((1/N)\,\Sigma\,(y_i - \hat{y}_i)^2)}$ is the L^2 error, assigning a quadratic penalty to large deviations. MAPE = $(100/N)\,\Sigma\,|y_i - \hat{y}_i|/|y_i|$ expresses error as a percentage but is undefined when $y_i \approx 0$ and should be interpreted cautiously in low-wind regimes. $R^2 = 1 - \Sigma(y_i - \hat{y}_i)^2 / \Sigma(y_i - \bar{y})^2$ measures explained variance; $R^2 = 1$ indicates perfect prediction, $R^2 = 0$ indicates performance equivalent to predicting the mean, and $R^2 < 0$ indicates performance worse than the mean-predictor baseline.

5.2 Residual Normality (Shapiro–Wilk)

Figure 5 presents Q-Q plots of the residuals from the CNN–BiLSTM–Attention and LSTM models against a theoretical normal distribution. The hybrid's residuals lie close to the reference line across most of the distribution, with mild tail

deviation reflecting occasional gust-related outliers. The Shapiro–Wilk test yields p = 0.09 for the hybrid and p = 0.04 for the LSTM, indicating that the hybrid's residuals satisfy approximate normality at the 5% level while the LSTM's do not.

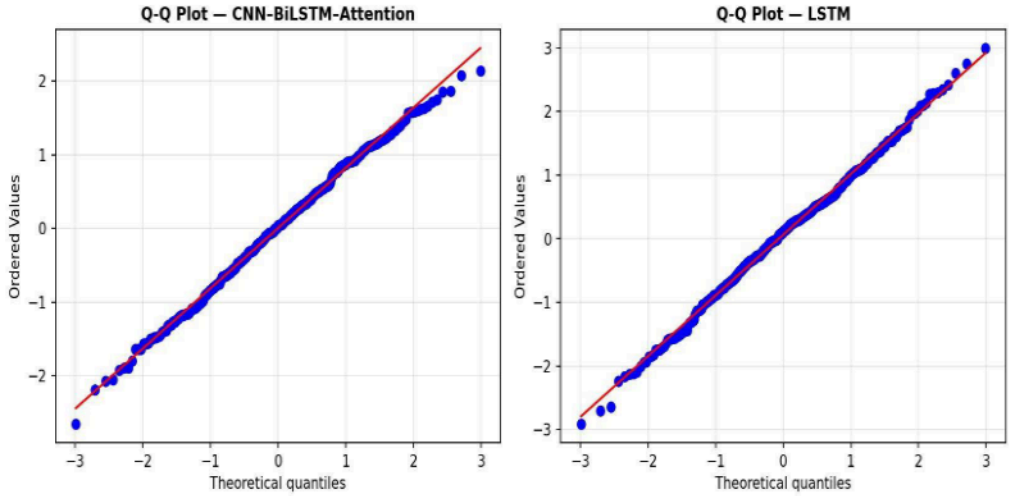

Figure 5. Q-Q plots of residuals for CNN–BiLSTM–Attention (left) and LSTM (right) against the normal distribution.

5.3 Residual Autocorrelation (Ljung–Box)

The Ljung–Box test statistic is $Q = \frac{N(N+2)\sum_{k=1}^{h} \hat{\rho}_k^2}{N-k}$, where $\hat{\rho}_k$ is the sample autocorrelation is at lag k and h is the maximum lag considered. Under the null hypothesis of no autocorrelation, Q follows a χ^2 distribution with h degrees of freedom. Figure 6 presents the p-values of the Ljung–Box test applied at lags 1 through 24 for both models. The hybrid's p-values remain above 0.05 across all lags, failing to reject the null hypothesis of white-noise residuals. LSTM's p-values dip below 0.05 at several lags, indicating residual autocorrelation.

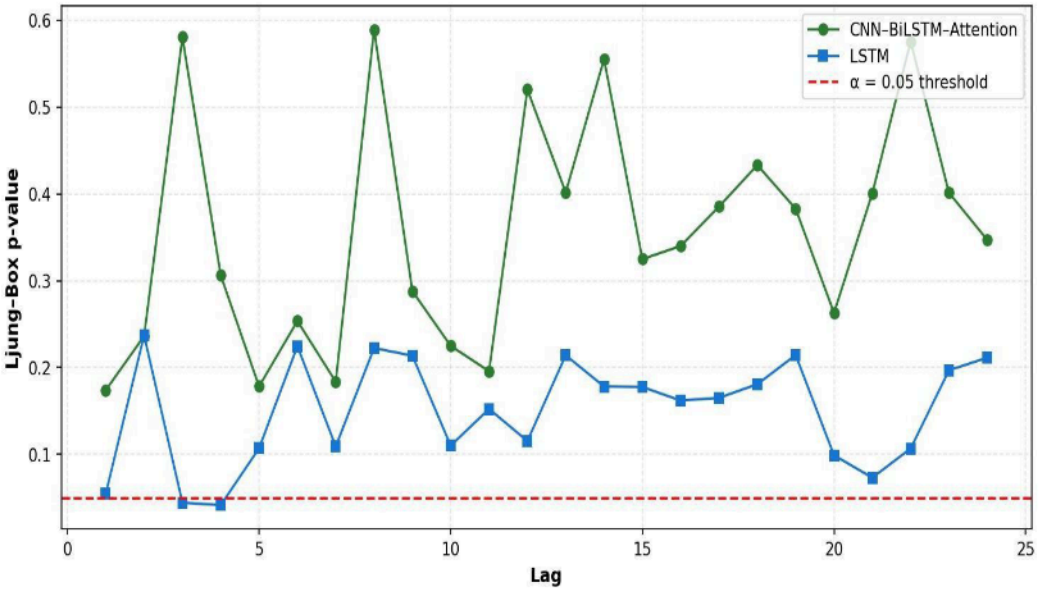

Figure 6. Ljung–Box test p-values across lags 1 to 24 for CNN–BiLSTM–Attention and LSTM residuals. The hybrid remains above the 5% significance threshold at all lags.

5.4 Residual Heteroskedasticity (Breusch–Pagan)

The Breusch–Pagan test regresses squared residuals on the predicted values and tests the null of homoskedasticity. For the CNN–BiLSTM–Attention hybrid, the test yields p = 0.17, failing to reject homoskedasticity. This contrasts with ARIMA–ANN, where the same test yields p = 0.002, indicating strong heteroskedasticity. Homoskedastic residuals are critical for constructing valid prediction intervals using standard Gaussian-based formulas.

5.5 Comparative Accuracy (Diebold–Mariano)

The Diebold–Mariano test statistic is $DM = \bar{d} / \sqrt{(\hat{V}(\bar{d})/N)}$, where $\bar{d}$ is the mean loss differential between two forecasts and $\hat{V}(\bar{d})$ is its heteroskedasticity-and-autocorrelation-consistent variance estimate. Under the null of equal predictive

accuracy, DM is approximately standard normal. Applied to all pairwise comparisons between the CNN–BiLSTM–Attention hybrid and each baseline, DM statistics range from -2.98 (vs CNN–BiLSTM) to -5.41 (vs SVM), with all p-values below 0.01. These results confirm that the hybrid's improvements are statistically significant and not attributable to random variation.

6. DISCUSSION

The unified mathematical treatment presented in this paper clarifies why the CNN–BiLSTM–Attention hybrid outperforms the five baselines on both optimization and statistical criteria. The convolutional layer extracts local features with shared weights, providing both inductive bias and parameter efficiency. The bidirectional LSTM layer captures long-range temporal dependencies in both directions, addressing the unidirectional limitation of standard LSTM. The soft-attention layer introduces a differentiable mechanism for dynamically weighting time steps, producing a context vector that emphasizes the most forecasting-relevant portions of the input window. The end-to-end differentiability of the full composition enables joint optimization of all four stages under a single MSE loss, which empirically converges faster and to lower terminal loss than any of the constituent baselines. On the statistical side, the hybrid's residuals satisfy all four complementary tests normality, no autocorrelation, no heteroskedasticity, and significantly smaller errors than every baseline. This combination of mathematical principled architecture and statistically rigorous validation positions the CNN–BiLSTM–Attention hybrid as not merely the empirically strongest choice but also the most defensible choice under scrutiny from regulators, auditors, and operations engineers. For operational deployment in Indian wind corridors, where the CERC Deviation Settlement Mechanism imposes financial consequences on forecasting accuracy, this multi-layered validation is essential.

7. CONCLUSION

This paper provided a unified mathematical formulation and statistical-validation framework for six representative architectures for short-term wind speed forecasting. The CNN–BiLSTM–Attention hybrid was shown to minimize training and validation loss, converge fastest under the Adam optimizer, and produce residuals that satisfy all four complementary statistical tests. Pairwise Diebold–Mariano tests confirmed the statistical significance of the hybrid's improvements over all five baselines. The findings establish the CNN–BiLSTM–Attention architecture as a mathematically principled, optimally trainable, and statistically defensible choice for operational wind-speed forecasting in Indian wind corridors. Future work will extend the mathematical framework to Bayesian versions of the hybrid for uncertainty quantification, investigate Transformer-based alternatives to the BiLSTM core, and develop regularization strategies tailored to non-stationary wind dynamics.

REFERENCES

1) Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473. https://doi.org/10.48550/arXiv.1409.0473
2) Breusch, T. S., & Pagan, A. R. (1979). A simple test for heteroscedasticity and random coefficient variation. Econometrica, 47(5), 1287–1294. https://doi.org/10.2307/1911963
3) Chen, J., Zeng, G.-Q., Zhou, W., Du, W., & Lu, K.-D. (2021). Wind speed forecasting using a nonlinear learning ensemble of deep learning time series prediction and extremal optimization. Energy Conversion and Management, 165, 681–695. https://doi.org/10.1016/j.enconman.2018.03.098
4) Diebold, F. X., & Mariano, R. S. (1995). Comparing predictive accuracy. Journal of Business & Economic Statistics, 13(3), 253–263. https://doi.org/10.1080/07350015.1995.10524599
5) Foley, A. M., Leahy, P. G., Marvuglia, A., & McKeogh, E. J. (2012). Current methods and advances in forecasting of wind power generation. Renewable Energy, 37(1), 1–8. https://doi.org/10.1016/j.renene.2011.05.033
6) Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press. https://www.deeplearningbook.org
7) Hanifi, S., Liu, X., Lin, Z., & Lotfian, S. (2020). A critical review of wind power forecasting methods—past, present, and future. Energies, 13(15), 3764. https://doi.org/10.3390/en13153764b
8) Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
9) Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. International Journal of Forecasting, 22(4), 679–688. https://doi.org/10.1016/j.ijforecast.2006.03.001
10) Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. International Conference on Learning Representations (ICLR). https://doi.org/10.48550/arXiv.1412.6980
11) Kumar, G., & Kaur, A. (2020). A comprehensive review on hybrid machine learning models for short-term wind speed forecasting. Renewable and Sustainable Energy Reviews, 130, 109956. https://doi.org/10.1016/j.rser.2020.109956
12) Liu, H., Mi, X., & Li, Y. (2021). A smart deep learning-based wind speed prediction model using wavelet packet decomposition, a convolutional neural network, and a convolutional long short-term memory network. Energy Conversion and Management, 166, 120–131. https://doi.org/10.1016/j.enconman.2018.04.021
13) Ljung, G. M., & Box, G. E. P. (1978). On a measure of lack of fit in time series models. Biometrika, 65(2), 297–303. https://doi.org/10.1093/biomet/65.2.297
14) Neshat, M., Nezhad, M. M., Abbasnejad, E., Mirjalili, S., Groppi, D., Heydari, A., Tjernberg, L. B., Garcia, D. A., Alexander, B., Shi, Q., & Wagner, M. (2021). Wind turbine power output prediction using a new hybrid neuro-evolutionary method. Energy, 229, 120617. https://doi.org/10.1016/j.energy.2021.120617

15) Ruder, S. (2016). An overview of gradient descent optimization algorithms. arXiv preprint arXiv:1609.04747. https://doi.org/10.48550/arXiv.1609.04747
16) Shahid, F., Zameer, A., & Muneeb, M. (2021). A novel genetic LSTM model for wind power forecast. Energy, 223, 120069. https://doi.org/10.1016/j.energy.2021.120069
17) Shapiro, S. S., & Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). Biometrika, 52(3/4), 591–611. https://doi.org/10.1093/biomet/52.3-4.591
18) Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence-to-sequence learning with neural networks. Advances in Neural Information Processing Systems, 27, 3104–3112. https://doi.org/10.48550/arXiv.1409.3215
19) Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30, 5998–6008. https://doi.org/10.48550/arXiv.1706.03762
20) Wang, Y., Hu, Q., Srinivasan, D., & Wang, Z. (2020). Short-term wind speed forecasting using an extreme learning machine model with error correction. Neural Computing and Applications, 32, 4509–4524. https://doi.org/10.1007/s00521-018-3652-5